Keeping MeshCore Firmware Updated

Keeping your MeshCore nodes on current firmware is important for stability, interoperability, and security. This page covers why updates matter, how to check your current version, update strategies for deployed infrastructure, and how to handle rollbacks.

Why Updates Matter

Bug Fixes

MeshCore is actively developed software. Each release typically resolves routing edge cases, BLE connectivity issues, memory leaks, and hardware-specific quirks. Running old firmware means running known bugs that may have already been fixed.

Performance Improvements

Routing algorithm refinements, radio parameter tuning, and message handling optimizations are regularly incorporated. A network of nodes all running the same recent firmware will generally route more efficiently than one running a mixture of old builds.

New Features

New capabilities - new sensor types, new room server features, new CLI commands, new position reporting formats - are only available in the firmware version that introduced them. Staying reasonably current ensures you can use new functionality as it becomes available.

Security Patches

While MeshCore is a mesh radio protocol rather than an internet-facing service, vulnerabilities can still exist. Malformed packet handling bugs, cryptographic implementation issues, and BLE pairing weaknesses are all possible attack surfaces. Security-relevant fixes are tagged in release notes;

apply them promptly.

Version Compatibility

MeshCore nodes on significantly different firmware versions may have interoperability limitations. Keeping your infrastructure nodes current minimizes the risk of incompatibility with nodes running newer client firmware.

Checking Your Current Firmware Version

There are two ways to check the firmware version on a node:

Via the MeshCore App

Connect to the node via the MeshCore app. Navigate to the node's detail or settings view. The firmware version is displayed in the device information section.

Via the MeshCore CLI

Connect to your node using a BLE serial terminal or the MeshCore CLI tool and run:

```
ver
```

This prints the node's firmware version. MeshCore firmware is currently in the 1.x series, so the output is an illustrative line such as:

```
v1.15.0
```

The `ver` command reports the firmware version. To see the hardware/board name, use the separate `board` command. For runtime health (battery, uptime, queue) use `stats-core`.

Update Strategy for Infrastructure Nodes

Repeaters and room servers are infrastructure - other users depend on them. Updating carelessly can cause network disruption. Follow this strategy:

1. Test on a Non-Critical Node First

If you operate multiple nodes, update one non-critical node (a spare, or the lowest-traffic repeater) to the new firmware first. Run it for 24 - 48 hours and verify:

- The node comes back online and connects to the mesh after the update.
- Routing works correctly through the node.
- No unexpected reboots or radio lockups occur.
- BLE connectivity from the app functions normally.

2. Preserve Configuration Before Updating

Before updating any node, record its current configuration:

- Node name
- Frequency preset and any custom radio parameters
- TX power setting
- Any custom channel configurations (for room servers: room name, password)

Use the CLI `get` queries (for example `get radio`, `get tx`) and `infos` to read back the current settings, and screenshot or copy the output. While configuration is generally preserved across firmware updates (stored in non-volatile flash separate from the firmware), a failed or interrupted flash can result in settings being wiped.

3. Update During Low-Traffic Periods

Infrastructure nodes go offline during flashing (typically 30 - 90 seconds). Schedule updates during periods when the network is least used to minimize impact on other users.

4. Update Infrastructure Before Clients

When a new major or minor version is released, update repeaters and room servers before client nodes. Infrastructure nodes carry traffic for all clients; having them on newer firmware ensures they can handle any new packet formats clients may start using.

How to Update

How you update depends on the board. ESP32 boards typically require USB flashing (the same process as initial flashing). nRF52 boards (RAK4631, T114, Seeed XIAO nRF52) additionally support over-the-air updates via the DFU app and the `start ota` CLI command, which avoids needing a USB connection. The USB web-flasher steps are:

1. Connect the node to a computer via USB.
2. Open the MeshCore Web Flasher at flasher.meshcore.io in Chrome or Edge.
3. Select your board type and firmware variant.
4. Select the new firmware version.
5. Click Connect, select the serial port, then click Flash.
6. Wait for the flash to complete and the board to reboot.
7. Verify the node is operational using `ver` (firmware version) and `stats-core` (battery/uptime/queue health).

For nRF52840 boards (RAK4631, T114, HT-n62): UF2 drag-and-drop is available as an alternative. Download the new `.uf2` file, enter bootloader mode (double-tap reset), and copy the file to the USB drive.

Rollback: Returning to a Previous Version

If a firmware update causes problems, you can return to any previous version:

1. Open the MeshCore Web Flasher.
2. Select your board and variant.
3. Use the **version selector** to choose the previous known-good version (older versions are retained in the flasher's version history).
4. Flash as normal.

For UF2 boards: download the previous version's `.uf2` file from the MeshCore GitHub releases page and flash it via drag-and-drop.

Note: Configuration is generally preserved across rollbacks. However, if a newer firmware version introduced a new configuration key that older firmware does not understand, the old firmware may ignore or reset that setting.

Coordinating Community Network Updates

If you operate nodes on a shared community network, coordinate updates with other network operators:

- Announce planned updates in your community's Discord, forum, or group chat before updating shared infrastructure.
- Share the release notes link so other operators can review what has changed.
- If a major version update is involved, agree on a migration window so all infrastructure nodes are updated together, minimizing the period of mixed-version operation.
- After updating, post a confirmation in the coordination channel so others know the node is back online and on the new version.

Same Version Compatibility Notes

Within the same major version, MeshCore nodes running different minor versions can generally communicate. However:

- As expected behavior for a forwarding mesh, a node running a minor version that introduced a new packet type may generate packets that older minor-version nodes do not fully process - they will typically still forward them but may not display them correctly. Check the release notes for any such changes.
- Patch releases within the same minor version are intended to be bug-fix-only and are generally interoperable, but MeshCore does not publish a formal compatibility guarantee - test before relying on mixed-version meshes.
- When in doubt, check the release notes for any compatibility warnings. The MeshCore team typically calls out cross-version compatibility issues explicitly.

Revision #5
Created 2026-05-03 06:02:51 UTC by Mesh America Admin
Updated 2026-06-09 14:39:21 UTC by Mesh America Admin